Sensing Vehicle Dynamics for Determining Driver Phone Use

Mobisys 2013

Stevens Institute of Technology

Oakland University

Rutgers University

Motivation

- Distracted driving due to mobile devices contributes to nearly one thousand fatalities per year
  - National Transportation Safety Board has called for a nationwide ban on mobile devices behind the wheel
  - The AT&T DriveSafe app, for example, silences the phone for incoming text messages while in driving mode
- Such approaches depend on the phone being able to sense when the user is driving
  - **Users are not very reliable at setting the status manually**

Contents

- Related Work
- System Design
- System Implementation
- Evaluation in real-road driving environments
- Discussion
- Conclusion

Related Work

- In the authors' prior work
  - Exploits built-in Bluetooth hands-free systems in vehicles for audio localization of the phone
- Certain app requires special devices installed inside the vehicle
  - To enable blocking cellular communications of a specific phone, or even rely on a radio jammer
- Detection of the moving vehicle
  - Based on the embedded smartphone sensors and the cellular signal (e.g., accelerations and cellular signal strength)

Related Work (cont'd)

- Other studies use smartphone embedded sensors
  - To alert dangerous vehicle maneuvers
    - By utilizing the accelerometer and the orientation sensor
  - To monitor road conditions
    - E.g., the accelerometer is able to detect a large acceleration perpendicular to the road surface when the vehicle passes bumps or potholes on the road
  - To detect traffic accidents
    - By using the accelerometer and microphone
- Driver detection system (DDS)
  - Utilizing multiple sensors to capture the feature of driver's movement
    - (including accelerometer, gyroscope, and micro-phone)
  - Sensitive to the behavior of each individual

System Design

- Background
  - $a = wv = w^2 r = \frac{v^2}{r}$
    - Centripetal acceleration (a)
    - Angular speed (w)
    - Tangential velocity (v)
    - Radius (r)

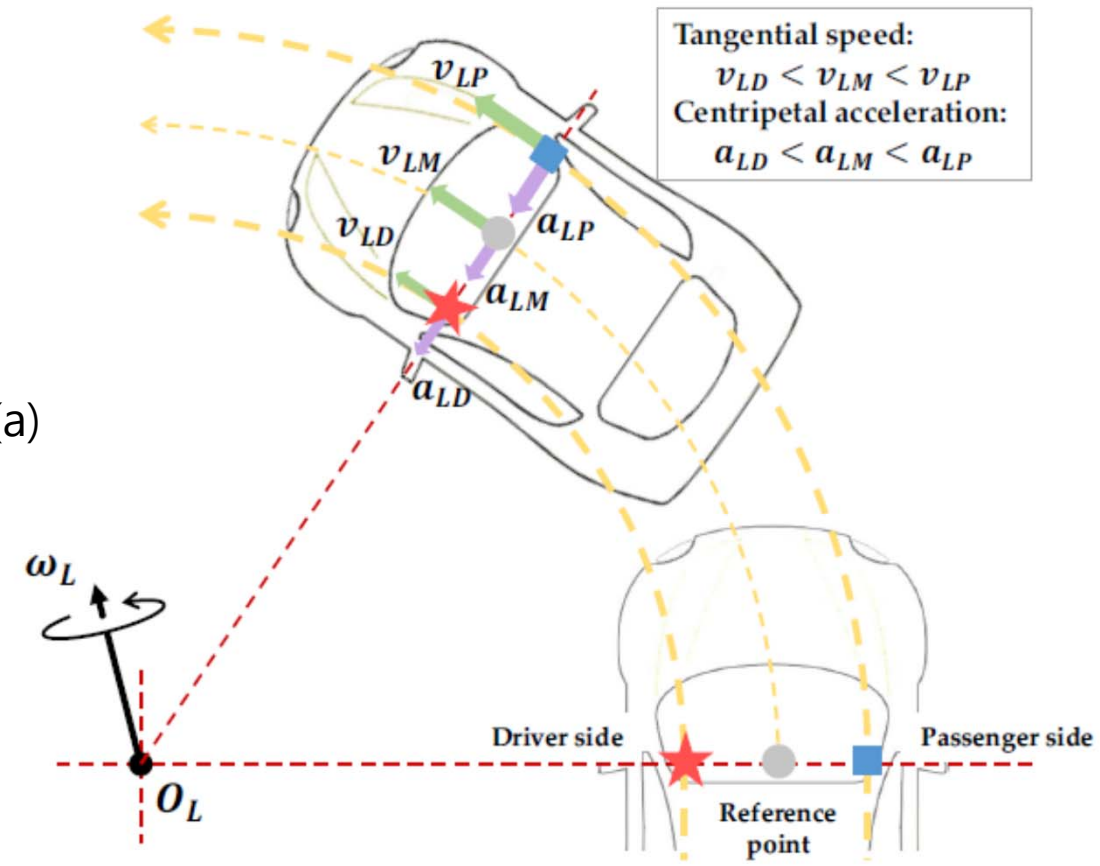


Figure 2: Illustration of different centripetal accelerations at different in-vehicle positions.

System Design (cont'd)

- Challenges and Goals
  - **Robustness to Real-Road Driving Environments**
    - The algorithm to obtain the centripetal acceleration has to be robust to deal with real road driving environments (e.g., running engine or wind contributing to noisy sensor readings)
  - **Achieving Single Phone Sensing**
    - The approach should work even when only a single phone is present
  - **Determining the Pose of the Phone**
    - The measured sensor readings can be used with knowing the pose of the phone inside the vehicle
  - **Computational Feasibility on Smartphones**

System Design (cont'd)

- System Overview
  - Basic idea is to examine the centripetal acceleration
  - Three ways of utilizing a reference centripetal acceleration
    - A low-cost cigarette lighter adapter containing an accelerometer
    - Usage of the speed of the vehicle obtained from the OBD-II
    - A passenger phone
      - Rely on the presence of a second phone

System Design (cont'd)

- System Overview (cont'd)
  - Three sub-tasks
    - Coordinate Alignment
      - The centripetal acceleration and angular speed derived from the phone sensors are aligned with the vehicle's
      - When the system starts or it detects the gyroscope readings crossing certain thresholds, which is caused by the change of phone's position
    - Data Calibration
      - Aims to synchronize the traces from different sources and reduce the hardware bias caused by different phone models
    - Position Detection
      - Determines the phone's position in car leveraging the cumulative difference of centripetal acceleration

System Design (cont'd)

- Sensing Vehicle Dynamics
  - **Phone and Vehicle Alignment**
    - Utilize the 3-axis accelerometer and 3-axis gyroscope
    - Two coordinate systems
      - $(\{X_p, Y_p, Z_p\})$ for the smartphone
      - $(\{X_c, Y_c, Z_c\})$ for the vehicle
    - Assume the smartphone's coordinate system is already aligned with the vehicle's
  - **Deriving Centripetal Acceleration via Accelerometers**
    - The X-axis acceleration will be positive (right turn) or negative (left turn)
  - **Determining Turn Directions using the Gyroscope**
    - The Z-axis gyroscope reading will be positive (left turn) or negative (right turn)

System Design (cont'd)

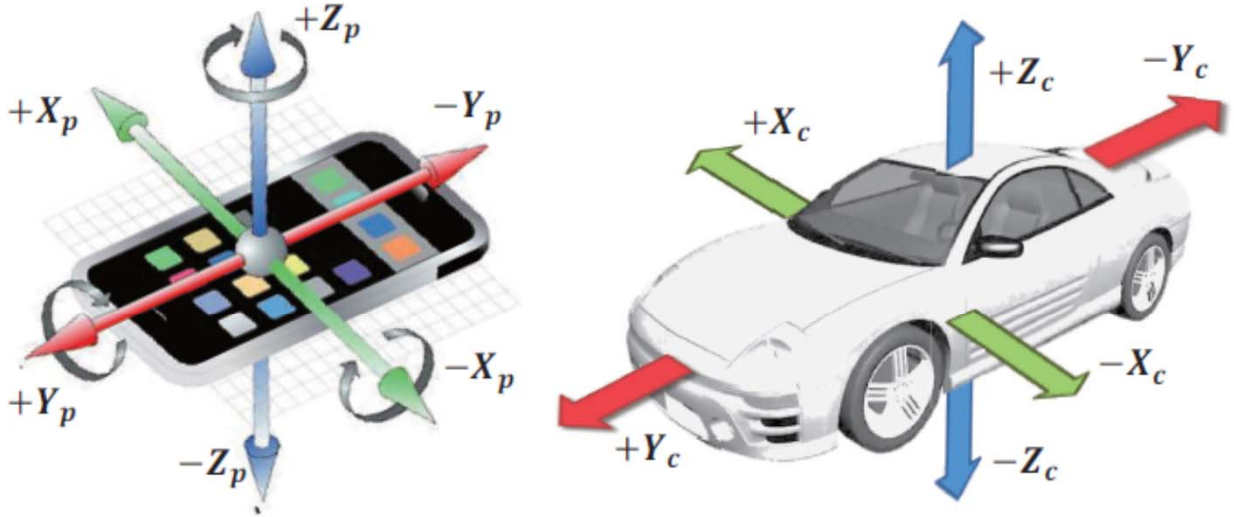
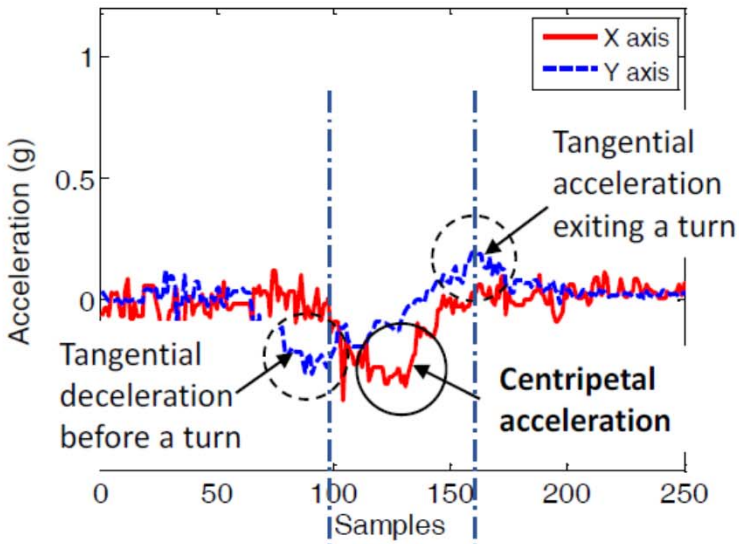


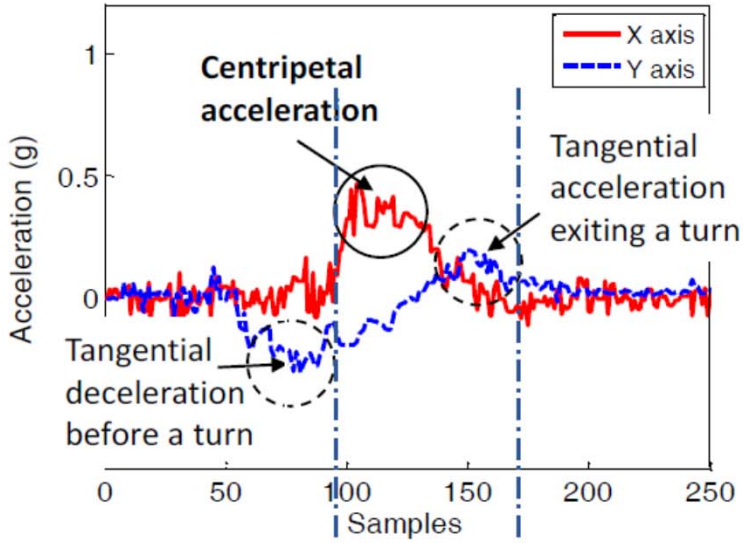
Figure 4: Coordinate systems of a smartphone and a vehicle.

System Design (cont'd)

- Sensing Vehicle Dynamics
  - **Phone and Vehicle Alignment**
    - Utilize the 3-axis accelerometer and 3-axis gyroscope
    - Two coordinate systems
      - $(\{X_p, Y_p, Z_p\})$ for the smartphone
      - $(\{X_c, Y_c, Z_c\})$ for the vehicle
    - Assume the smartphone's coordinate system is already aligned with the vehicle's
  - **Deriving Centripetal Acceleration via Accelerometers**
    - The X-axis acceleration will be positive (right turn) or negative (left turn)
  - **Determining Turn Directions using the Gyroscope**
    - The Z-axis gyroscope reading will be positive (left turn) or negative (right turn)



Left turns



Right turns

System Design (cont'd)

- Algorithm for Sensing Driver Phone Use
  - $(a - a_M)\omega > 0$: passenger phone
  - $(a - a_M)\omega < 0$: driver phone
  - a: the centripetal acceleration of the smartphone measured from its X-axis accelerometer
  - a_M: the centripetal acceleration of the reference position
  - ω: the angular speed measured from smartphone's Z-axis gyroscope sensor, reflecting the turn direction (e.g., $\omega > 0, left\ turn$)

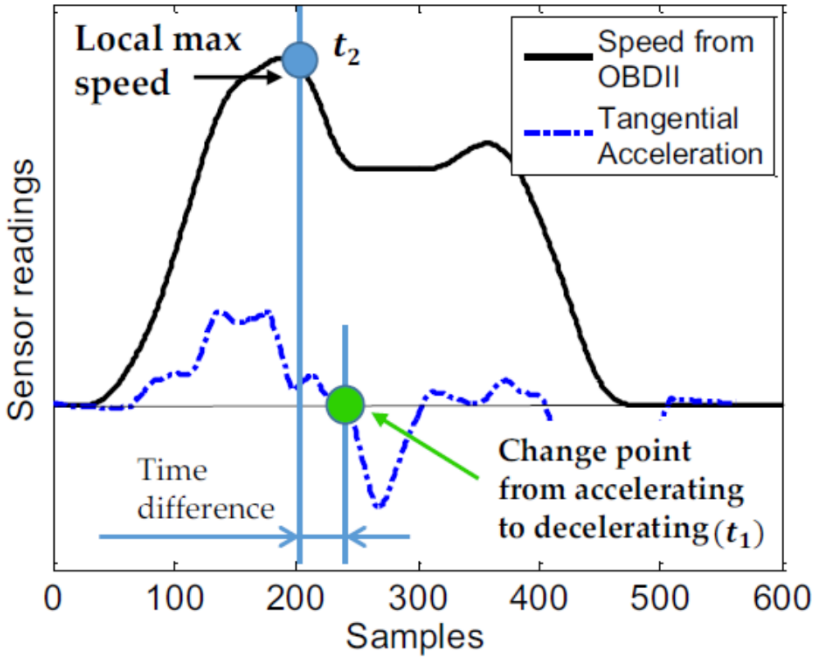
System Implementation

- Data Calibration for Enhanced Reliability
  - Many factors (such as running engines and wind) affect the readings from the accelerometers and gyroscopes on smartphones
    - Filter noise from sensor readings (data interpolation)
    - Ensure the synchronization between sensor readings from different sources (trace synchronization)
    - Reduce bias caused by hardware difference in smartphones (acceleration adjustment)
  - **Data Interpolation**
    - To reduce the noise, apply a moving average filter
    - $[t_0, t_0 + \delta, t_0 + 2\delta, \dots]$ $where\ \delta: the\ interpolation\ step$, $t_0: the\ starting\ time\ stamps$
      - The real sampling interval has a small variation even with a fixed sampling rate

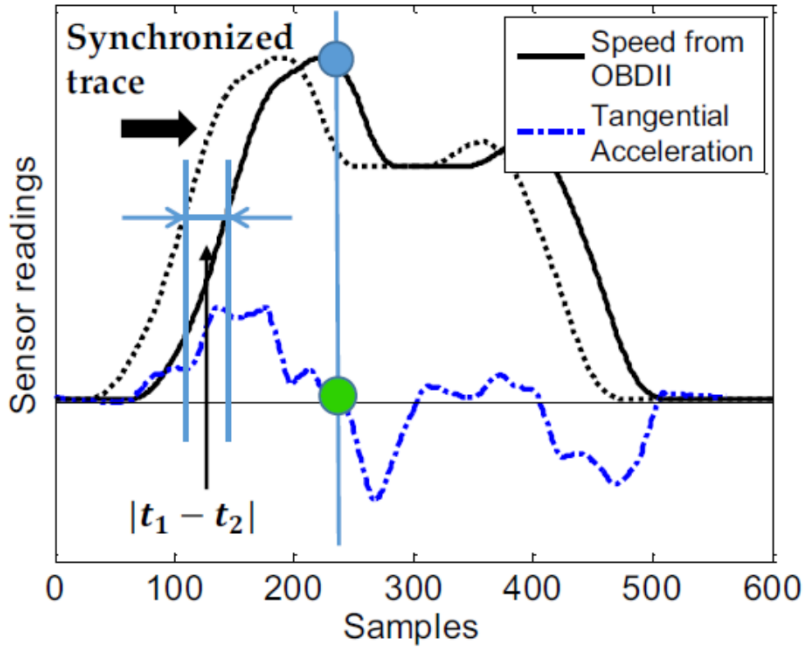
System Implementation (cont'd)

- Data Calibration for Enhanced Reliability (cont'd)
  - **Trace Synchronization**
    - Synchronize the sensor readings from
      - The phone
      - The reference point (e.g., the cigarette lighter adapter or OBD-II port adapter)
    - These two sequences of readings are synchronized
      - When the cross correlation between theses two sequence of readings in time series
      - Because both sequences reflect vehicle's movement
    - Synchronization becomes more challenging using the speed obtained from ODB-II port adapter
      - The time point that the vehicle changes from acceleration to deceleration during normal driving is the point that the vehicle reaches its maximum speed

System Implementation (cont'd)



(a) Before synchronization



(b) After synchronization

Illustration of trace synchronization mechanism via tangential acceleration

System Implementation (cont'd)

- Data Calibration for Enhanced Reliability (cont'd)
  - **Acceleration Adjustment**
    - To reduce the bias caused by hardware differences
    - The readings on the X-axis accelerometer of the phone should be zero
      - When the vehicle is moving along a straight line
    - Nevertheless, the acceleration on the X-axis may have a constant value
    - Adjustment
      - Determine the time period that the vehicle is driving along a straight line (i.e., the time period with no rotation rate on the Z-axis gyroscope)
      - Calculate the mean value of the X-axis acceleration during this time period
      - Subtract the calculated mean value form all the X-axis acceleration readings

System Implementation (cont'd)

- Coordinate Alignment
  - Aim to find a rotation matrix R
    - To rotate the phone's coordinate system to match with the vehicle's
    - $(\{X_p, Y_p, Z_p\})$ the phone's coordinate system
    - $(\{X_c, Y_c, Z_c\})$ the vehicle's coordinate system
      - Unit coordinate vectors: $\hat{\imath}, \hat{\jmath}, \hat{k}\ for\ X_c, Y_c\ and\ Z_c\ axis\ respectively$
        - $\hat{\imath} = [1, 0, 0]^T$
    - Utilize smartphone's accelerometer and gyroscope readings

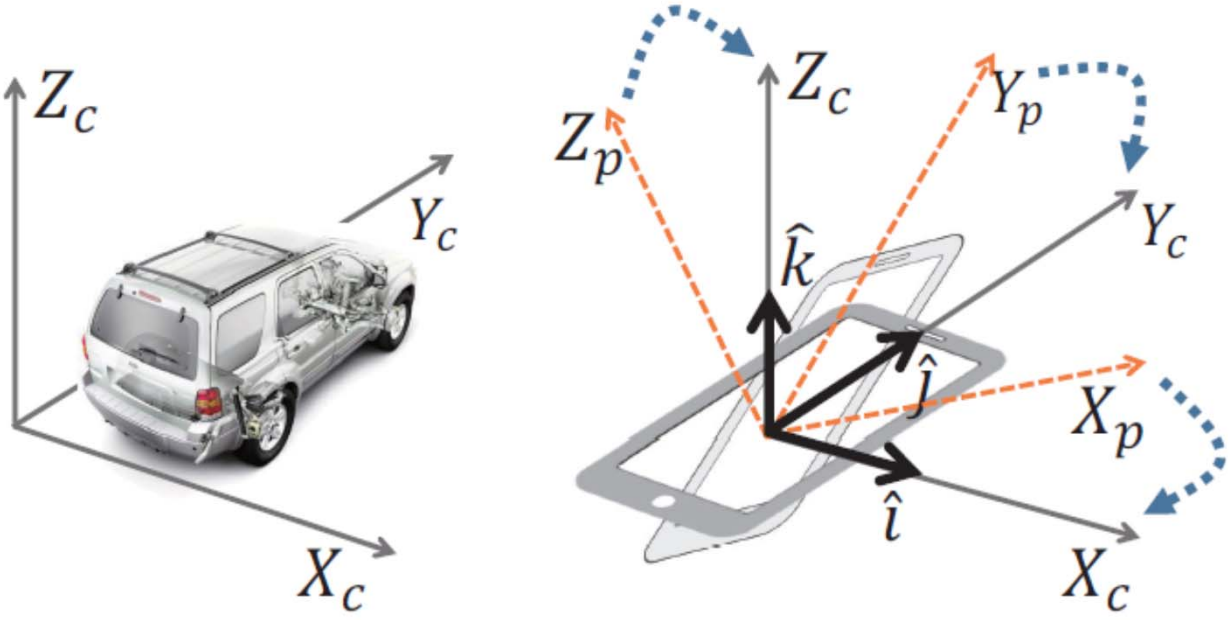


Figure 8: Illustration of how the phone's coordinate system is aligned to the vehicle's coordinate system.

Evaluation in Real-Road Driving Environments

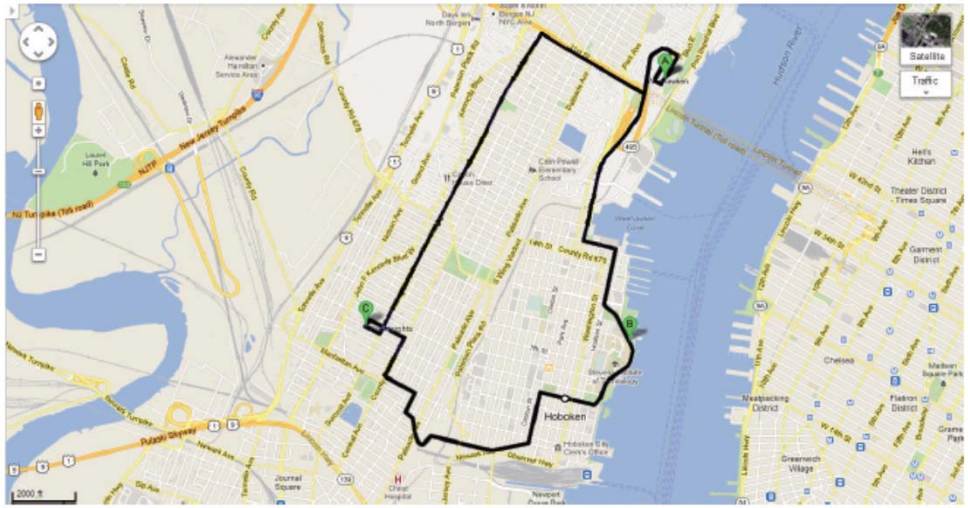
- Experimental Setup
  - **Phones and Vehicles**
    - iphone4 and HTC 3D
    - Both have a Bluetooth radio, 3-axis accelerometer and gyroscope
    - iphone4: 1GHz ARM Cortex-A8 CPU, 512M RAM running with iOS 5.2
    - HTC 3D: Qualcomm MSM8660 1.2GHz CPU and 1G RAM running with Android 2.4
    - Both the accelerometer and gyroscope sampling rate: 20 samples/s
    - Honda Accord (Car A) and Acura sedan (Car B)

Evaluation in Real-Road Driving Environments (cont'd)

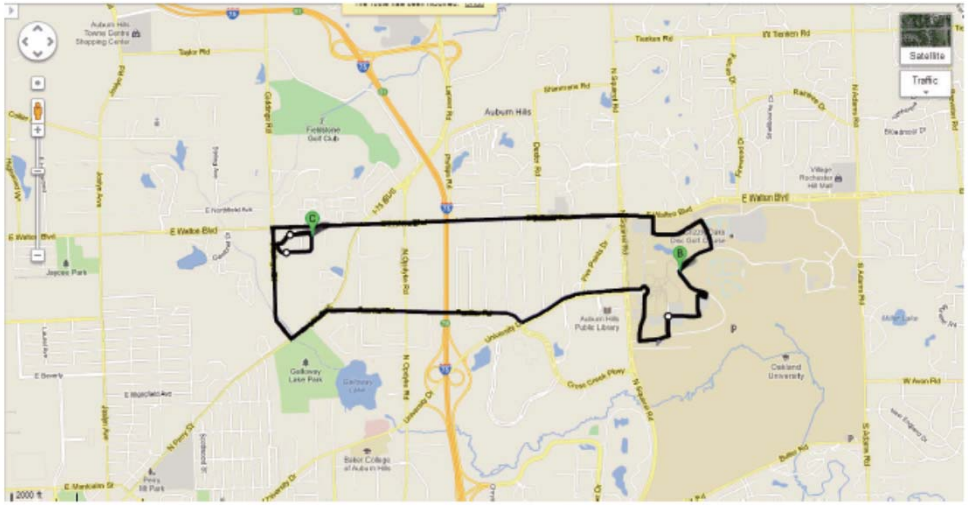
- Experimental Setup (cont'd)
  - **Real Road Driving Scenarios**
    - Typical urban setting
      - iphone 4 for over one month when Car A used as the daily commute vehicle in Hoboken, NJ
    - Typical suburban setting
      - HTC 3D for over one week when Car B used as the daily commute vehicle in Pontiac, MI
    - The phones are placed in either the driver or passenger seat/door
    - Each of the traces contains 10 to 20-minute of driving

Location	Left turns	Right turns	Car	Phone
Hoboken,NJ	292	278	Car A	iphone4
Pontiac,MI	211	219	Car B	HTC 3D

Evaluation in Real-Road Driving Environments (cont'd)



(a) Driving trajectory in Hoboken, NJ



(b) Driving trajectory in Pontiac, MI

Daily commute routes used for real-road driving evaluation in Hoboken, NJ and Pontiac, MI

Evaluation in Real-Road Driving Environments (cont'd)

- Experimental Setup (cont'd)
  - **Prototype**
    - Runs as an Android App
    - Collects readings from the accelerometer and gyroscope in the smartphone
    - Runs through the detection algorithm
    - Works with the OBD-II port adapter via Bluetooth
    - Present the results using iPhone4 by applying trace driven off-line analysis

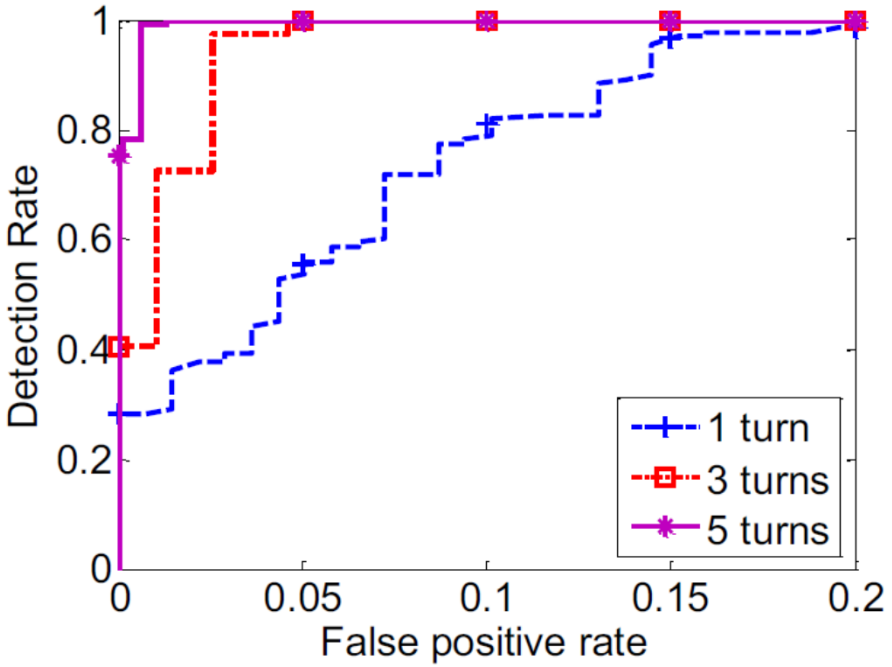
Evaluation in Real-Road Driving Environments (cont'd)

- Experimental Setup (cont'd)
  - **Metrics**
    - Accuracy
      - The percentage of the trails correctly classified as driver or passenger phone use
    - Detection Rate (DR)
      - The percentage of driver phone use correctly identified
    - False Positive Rate (FPR)
      - The percentage of passenger phone use classified as driver phone use
    - Detection Latency
      - The time needed to make a decision on whether it's a driver phone use or passenger use starting from driving a vehicle

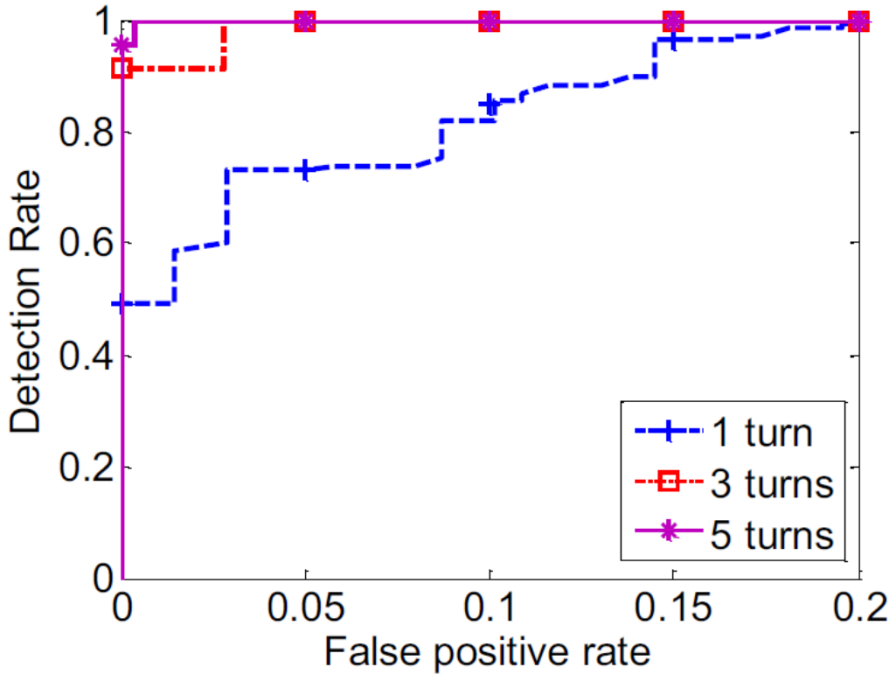
Evaluation in Real-Road Driving Environments (cont'd)

- Evaluation Using a Cigarette Lighter Adapter
  - 80% detection rate with less than a 10% false positive rate for both traces with in one turn
  - By utilizing multiple turns
    - 97% (detection rate), 3% (false positive rate) with 3 turns in Hoboken
    - 99.1% (detection rate), 1% (0.3% to be exact in Hoboken, false positive rate) with 5 turns
  - Indicates the system can achieve very high detection accuracy with only a few turns

Evaluation in Real-Road Driving Environments (cont'd)



(a) Car A, iPhone4, Hoboken, NJ



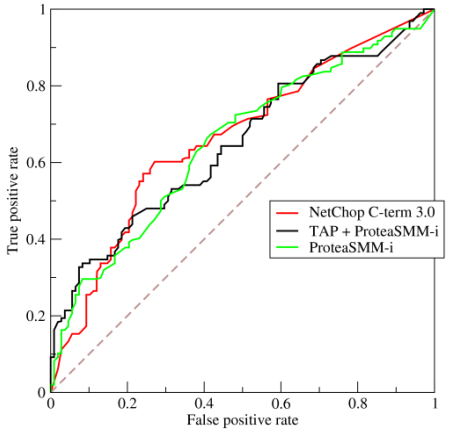
(b) Car B, HTC 3D, Pontiac, MI

ROC curve when using the adapter phone as the reference point

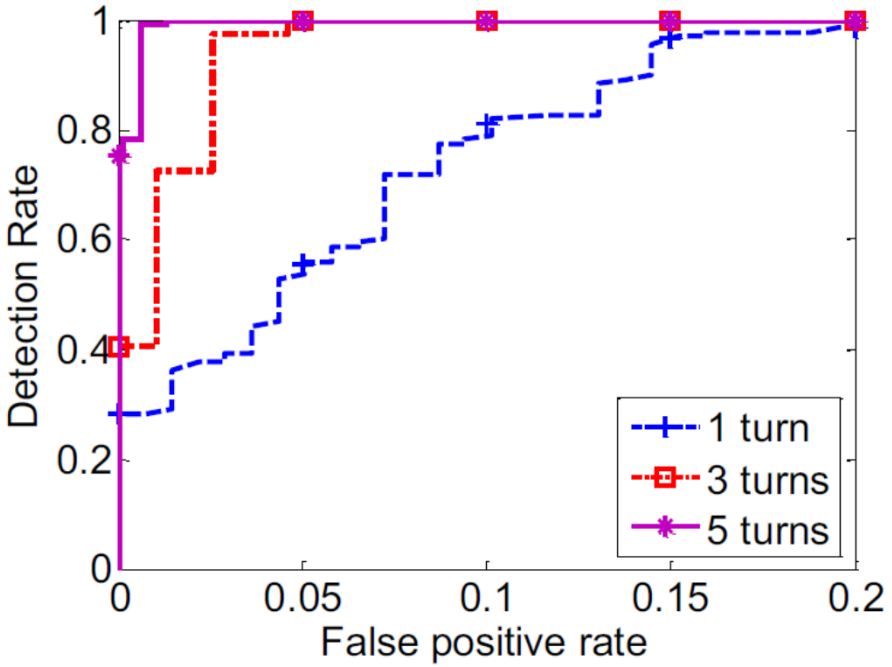
Receiver Operating Characteristic (ROC)

- In signal detection theory
  - ROC curve is a graphical plot which illustrates the performance of a binary classifier system as its discrimination threshold is varied

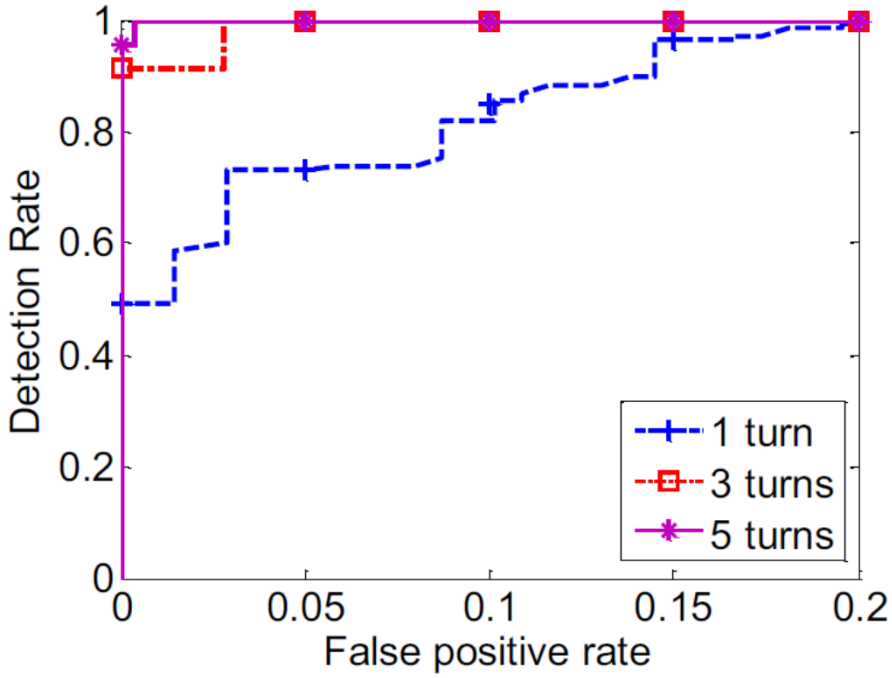
		actual value		
		p	*n*	total
prediction outcome	*p'*	True Positive	False Positive	P'
	n'	False Negative	True Negative	N'
	total	P	N	



Evaluation in Real-Road Driving Environments (cont'd)



(a) Car A, iPhone4, Hoboken, NJ



(b) Car B, HTC 3D, Pontiac, MI

ROC curve when using the adapter phone as the reference point

Evaluation in Real-Road Driving Environments (cont'd)

- Evaluation Using an OBD-II Port Adapter
  - **Filtering Turns with Angular Speed**
    - The larger angular speed, the more powerful discrimination in the centripetal acceleration
    - Filtering out small angular speed will improve the detection performance
    - 80% of the turns > 0.5 rad/s
  - **Results**
    - 20% better under turn selection
    - 80% (detection rate), 20% (false positive rate) under only 1 set of mixed turns
    - 91% (detection rate), 5% (false positive rate) based on 3 sets of mixed turns

System Design

- Background
  - $a = wv = w^2 r = \frac{v^2}{r}$
    - Centripetal acceleration (a)
    - Angular speed (w)
    - Tangential velocity (v)
    - Radius (r)

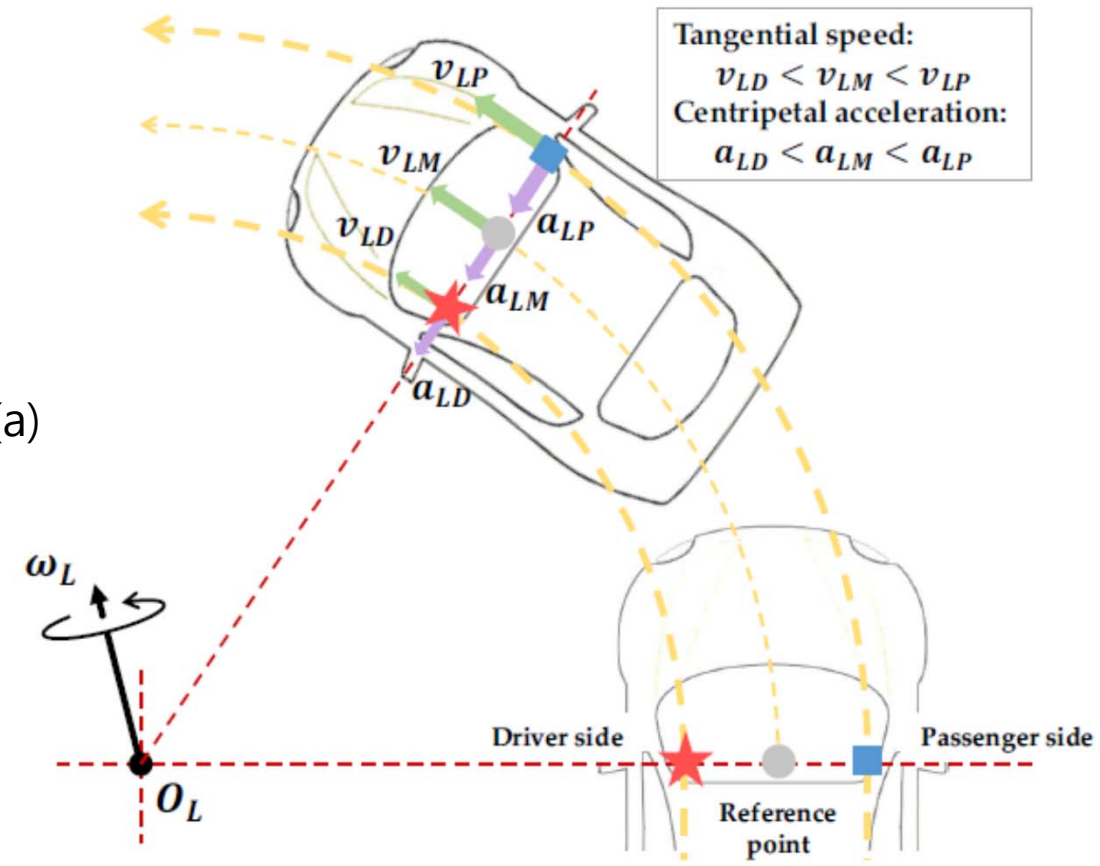
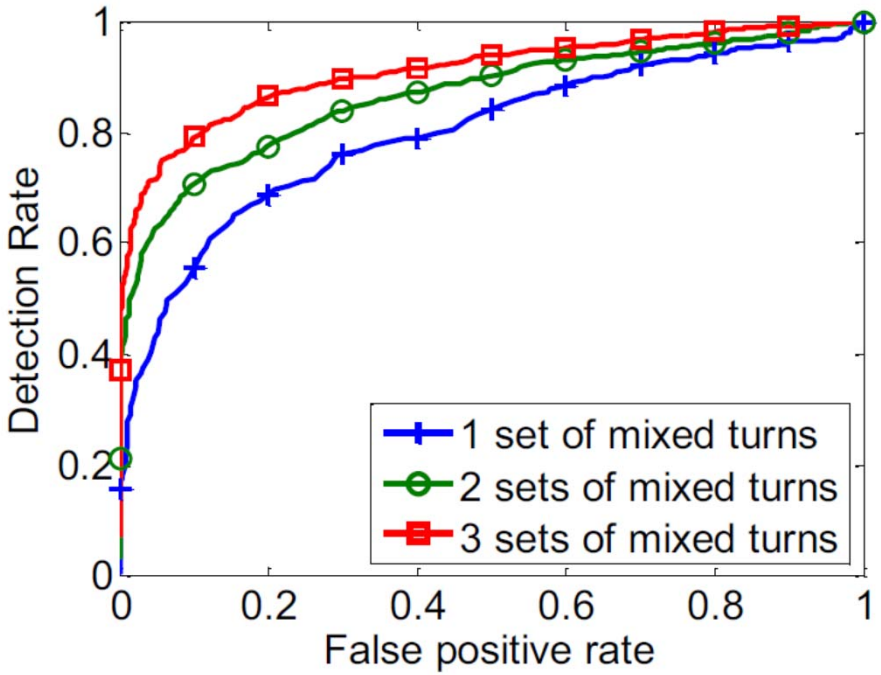
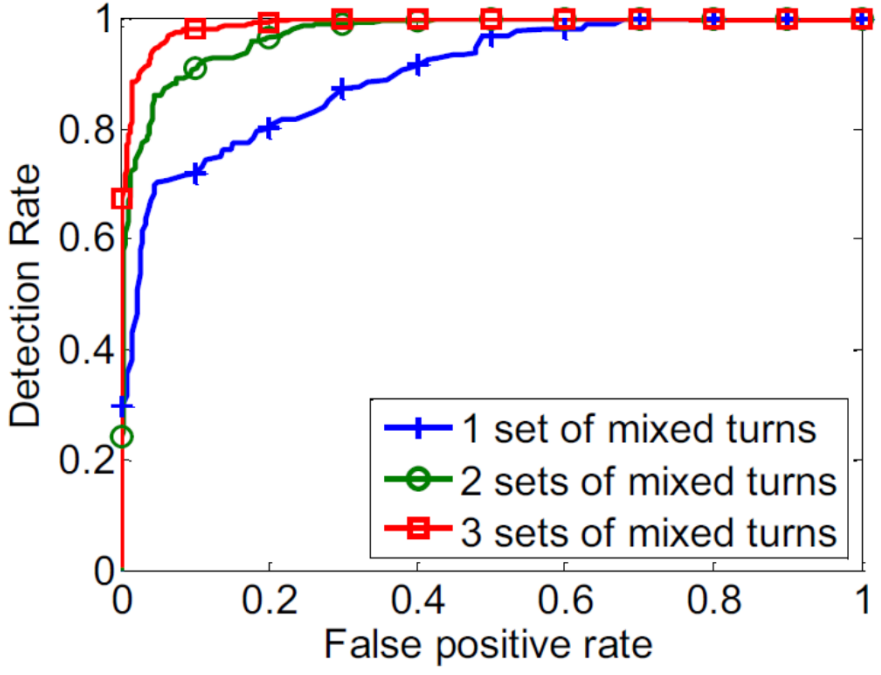


Figure 2: Illustration of different centripetal accelerations at different in-vehicle positions.

Evaluation in Real-Road Driving Environments (cont'd)



(a) Without turn selection



(b) With turn selection

ROC curve when using OBD-II port adapter as the reference point under mixed turns using iPhone4, Car A in Hoboken NJ.

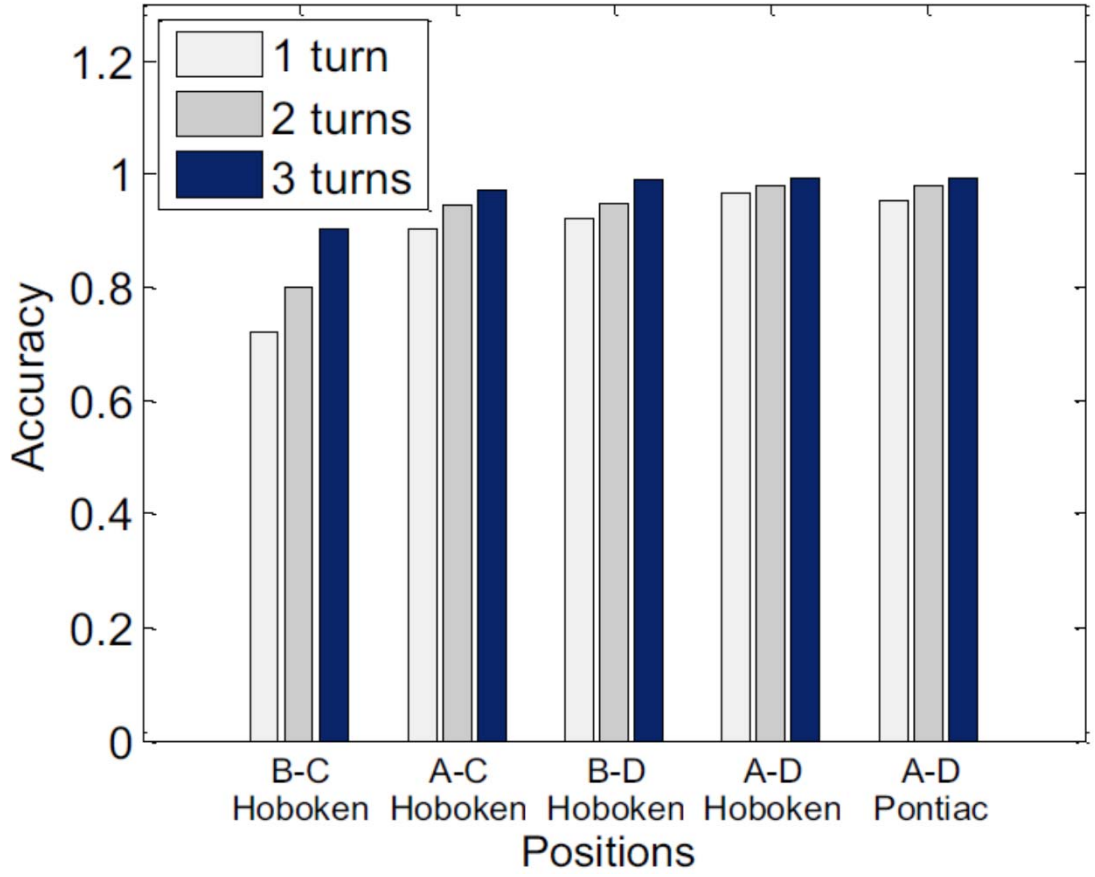
Evaluation in Real-Road Driving Environments (cont'd)

- Evaluation Using Dual Phones
  - 38% of automobile trips include passengers
  - Position A and B: driver's left and right pocket, respectively
  - Position C and D: passenger's left and right pocket, respectively



A B C D

Evaluation in Real-Road Driving Environments (cont'd)



Evaluation in Real-Road Driving Environments (cont'd)

- Detection Latency
  - Algorithm execution time (level of sub-millisecond) + turning time
  - The time interval between two turns (measured between two maximum angular speeds)
    - 28s in Hoboken, NJ
      - The urban city setting makes the driving routes involve more traffic lights and stop signs
    - 18s in Pontiac, MI

Discussion

- Extended System with Front-Rear Detection
  - Basic idea
    - The acceleration forces on a vehicle when passing over speed bumps, potholes, or other uneven surfaces are also position dependent
    - Addressed in the author's previous work
      - 90% accuracy when passing two bumps
      - 94% accuracy when passing three bumps
- Towards Infrastructure-Free Driver Phone Use Detection
  - Using GPS speed instead of OBD-II speed
    - GPS speed is not sensitive to the in-vehicle position
- Integration with Additional Vehicle Sensors
  - Steering wheel angle measured by a steering wheel position sensor

Conclusion

- Demonstrate a low-infrastructure approach for discriminating between a phone in the driver or passenger position
  - It does not rely on a built-in hands-free Bluetooth system
  - The centripetal acceleration varies depending on the position
- Demonstrate the generality of the approach through extensive experiments
  - With two different phone types
  - With two different cars in the two cities
- Detection is often possible by the time a vehicle leaves a parking lot or before it reaches a main road, so the determination is available for the vast majority of trips